Real Time Facial Expression Recognition Using Local Binary Patterns and Linear Programming

Xiaoyi Feng[1], Jie Cui[1], Matti Pietikäinen[2], and Abdenour Hadid[2]

[1] College of Electronics and Information, Northwestern Polytechnic University, 710072 Xi'an, China
{fengxiao, cuijie}@nwpu.edu.cn
[2] Machine Vision Group, Infotech Oulu and Dept. of Electrical and Information Engineering,
P. O. Box 4500 Fin-90014 University of Oulu, Finland
{mkp, hadid }@ee.oulu.fi

Abstract. In this paper, a fully automatic, real-time system is proposed to recognize seven basic facial expressions (angry, disgust, fear, happiness, neutral, sadness and surprise). First, faces are located and normalized based on an illumination insensitive skin model and face segmentation; then, the Local Binary Patterns (LBP) techniques, which are invariant to monotonic grey level changes, are used for facial feature extraction; finally, the Linear Programming (LP) technique is employed to classify seven facial expressions. Theoretical analysis and experimental results show that the proposed system performs well in some degree of illumination changes and head rotations.

1 Introduction

Real-time facial expression recognition plays an important role in real applications such as human-computer interaction, telecommunication and psychological research etc. However, it is a challenging problem in the computer vision literature. In the past years, only few works have addressed this issue [1-5].

Michel and R.E. Kaliouby [1] employed a feature displacement approach for expression recognition. 22 facial features were extracted from the video stream and the displacement for each feature between a neutral and a representative frame of an expression were calculated and then input to SVM classifier for training or testing. In Kotsia and Pitas' system [2], the Candide grid nodes on the face were manually placed at the first frame of one image sequence, the distance of each node's coordinates between the first and the last frame of the image sequence was used as an input to a multi-class SVM system for expression classification. Anderson and Mcowan [3] located face in scene with face tracker, then determined motion of face region only using optical flow algorithm, at last they input motion data into neural networks and determine emotion. Park [4] presented a point-wise energy based method for expression recognition. At the analysis step, they computed the motion energies of facial features and extract the dominant facial features related to each expression. At the recognition step, they performed rule-based facial expression recognition on arbitrary images using the results of analysis. Zhou et al. [5] proposed a novel

A. Gelbukh, A. de Albornoz, and H. Terashima (Eds.): MICAI 2005, LNAI 3789, pp. 328 – 336, 2005.

network structure and parameters learning algorithm for embedded HMM based on AdaBoost and then used this optimized embedded HMM to real time facial expression recognition.

In this paper, we propose a novel, fully automatic and real time system for facial expression recognition. First, face is detected using a skin model and eyes are located based on the combination of face's geometrical structure and face segmentation; Then, the Local Binary Pattern (LBP) operator is used to efficiently describe facial expressions; Finally, the Linear Programming (LP) technique is used to classify seven basic expressions. The proposed system has no constraint on the first frame (the first frame may be with other expressions than neutral) and works well in some extent of illumination changes and head movement.

The rest of the paper is organized as follows. Face preprocessing is described in section 2 and feature extraction method is presented in section 3. In section 4, we introduce the classification method. Experimental results are described in section 5. Finally in section 6 we conclude the paper.

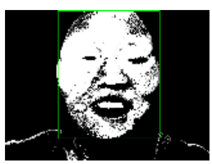
2 Face Preprocessing

Face preprocessing procedure includes three steps: face detection, eyes location and face normalization.

2.1 Face Detection with a Skin Locus

Martinkauppi et al. had found the Normalized Color Coordinates (NCC) combined with the skin locus most appropriate for skin detection under varying illumination [9]. To detect face-like area, the image presented in RGB color space is converted to the NCC space r, g and b. if r and b of a pixel fall into the area of the skin locus, the pixel belongs to skin. An example for skin detection is shown in Fig. 1. Considering real application, we select the largest skin component and regard it as face area (see the part inside the green box in Fig. 1).

(a) Original image



(b) Detected face

Fig. 1. Face detection result

2.2 Face Segmentation

Based on the knowledge that facial features are darker than their surroundings, morphological valley detectors are usually used for eyes detection, while these feature detection methods are sensitive to illumination changes (See Fig.2).

(a) Grey image (b) Velly detection (c) Detected eyes

Fig. 2. Velly-based eyes detection

Here we propose a novel method for facial feature detection based on the color information, which is insensitive to illuminations. Based on the observation that eyes and eyebrows contain less and lips contain more red elements than the skin part, the color face region is converted to a grey level image named color-ratio image as follows,

$$f(x, y) = \min(255, {b \times 255}/{r}) \tag{1}$$

Here $f(x, y)$ is the grey value of a pixel in position (x, y) in the color-ratio image and r and b are two chromaticities in NCC space. The color-ratio image corresponding to the image in Fig.1(a) is shown in Fig.3(a).

(a) (b) (c) (d)

Fig. 3. Color-based eyes detection (a) Color-ratio image (b) Upper face segmentation (c) Lower face segmentation (d) Detected eyes

The upper part and the lower part of the color-ratio image are then segmented respectively, according to the rules of minimizing intra-class variance and the results are shown in Fig.3 (b) and Fig.3 (c).

2.3 Eyes Location and Tracking

After the possible facial features are detected, a similar method as proposed in [10] is applied to evaluate feature constellations, using a geometrical face model including eyes, eyebrows and mouth. Based on experiments we have modified the face model to make the tracking procedure accurate and fast.

We first select facial features locating at the upper half of face area as eyes candidates and evaluate each eye candidate pair as follows,

$$E_{eyepair} = 0.5\exp(-10(\frac{D_{eyes} - 0.4B_{width}}{D})^2) + 0.25\left|\theta_{eyeleft} + \theta_{eyeright} - 2\times\theta\right| \tag{2}$$

Here B_{width} is the width of face bounding. Let $D = 0.8B_{width}$. D_{eyes} is the real distance of an eye candidate pair. θ, $\theta_{eyeleft}$ and $\theta_{eyeright}$ indicate directions of base line (line passing through center of a eye candidate pair), left eye candidate and right eye candidate, respectively.

For each eye candidate pair, other facial features are searched for and evaluated.

$$E_{feature} = \exp(-10(\frac{d_{feature} - D_{feature}}{D})^2) \tag{3}$$

Where $features = \{mouth, eyebrows\}$, $d_{feature}$ and $D_{feature}$ are real distance and reference distance from features to base line.

The total evaluation value is a weighted sum of the values for each facial features. The weights for each pair of eyes, mouth, and eyebrows are 0.4, 0.3, 0.1 and 0.05, respectively. The constellation with the largest evaluation value is assumed to real facial features. Fig.2 (c) and Fig.3 (d) are results of eyes detection.

It should be pointed out that during eyes tracking procedure, the reference distances are replaced by corresponding real distances, which can be obtained from the just processed frame.

2.4 Face Normalization

Face normalization is based on the position of two eyes and the distance between them. After face normalization, eyes position and distance between two eyes are the same. Fig. 4 shows one face normalization result. The size of each normalized image is 150×128.

Fig.4. Normalized face

3 Face Feature Extraction

Fig.5 is an illustration of the basic LBP operator [11]. The original 3×3 neighbourhood at the left is thresholded by the value of the centre pixel, and a binary pattern code is produced. The LBP code of the centre pixel in the neighbourhood is obtained by converting the binary code into a decimal code. It is obviously that LBP is invariant to grey level changes.

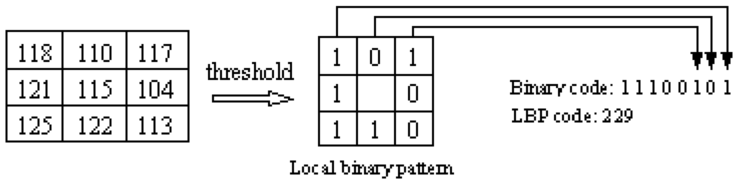


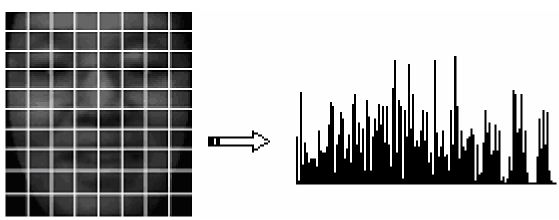
Fig. 5. The basic LBP operator

Based on this operator, each pixel of an image is labelled with an LBP code by thresholding its neighbourhood with the value of the centre pixel [11]. The 256-bin histogram of the labels (distribution of pattern codes) contains the density of each label over a local region, and can be used as a texture descriptor of the region.

The LBP operator, introduced by Ojala et al., has been shown to be a powerful measure of image texture. It has been applied to many problems with excellent performance [12, 13]. In our work, face images are seen as a composition of micro-patterns, which can be well described by LBP.

Now, feature extraction is implemented with the following steps:

1. **Divide the face image into small regions.** The size of each pre-processed image is 150×128. After experimenting with different block sizes, we choose to divide the image into 80 (10×8) non-overlapping blocks (See Fig.6 (a)).
2. **Calculate the LBP histogram from each region.** The LBP histogram of each region is obtained by scanning it with the LBP operator.
3. **Concatenate the LBP feature histograms into a single feature vector.** LBP histogram of each region is combined together to form a single feature vector representing the whole image (See Fig.6 (b)).



(a) Image blocks (b) LBP histogram

Fig. 6. An example of a facial image divided into 10×8 blocks

The idea behind using our approach for feature extraction is motivated by the fact that emotion is more often communicated by facial movement, which will change visible appearance. Our feature extraction method is capable of presenting facial appearances and so it can be used for representing facial expressions.

4 Expression Classification Based on Linear Programming

In [14, 15], a single linear programming (LP) formulation is proposed which generates a plane that minimizes an average sum of misclassified points belonging to two disjoint points set. We briefly describe this LP formulation below.

Consider the two point-sets A and B in the n-dimensional real space R^n represented by the $m \times n$ matrix A and the $k \times n$ matrix B, respectively. The separating plane is as follows:

$$P := \{x \mid x \in R^n, x^T \omega = \gamma\} \tag{4}$$

Here $\omega \in R^n$ is normal to the separating plane with a distance $\frac{|\gamma|}{\|\omega\|}$ to the origin.

The separating plane P determines two open half-spaces, $\{x \mid x \in R^n, x^T \omega > \gamma\}$ containing mostly points belong to A, and $\{x \mid x \in R^n, x^T \omega < \gamma\}$ containing mostly points belonging to B. That is we wish to satisfy

$$A\omega > e\gamma, \quad B\omega < \gamma \tag{5}$$

Here e is a vector of all 1s with appropriate dimension. To the extent possible, or upon normalization

$$A\omega \geq e\gamma, \; B\omega \leq e\gamma - e \tag{6}$$

Conditions (5) or (6) can be satisfied if and only if, A and B do not intersect, which in general is not the case. We thus attempt to satisfy (6) by minimizing some norm of the average violations of (6) such as

$$\min_{\omega,\gamma} \frac{1}{m}\|(-A\omega + e\gamma + e)_+\|_1 + \frac{1}{k}\|(B\omega - e\gamma + e)_+\|_1 \tag{7}$$

Here x_+ denotes the vector in R^n satisfying $(x_+)_i := \max\{x_i, 0\}, i = 1,2,...n$. The norm $\|.\|_p$ denotes the p norm, $1 \leq p \leq \infty$.

Formulation (7) is equivalent to the following robust linear programming formulation

$$\begin{aligned} &\min_{\omega,\gamma,y,z} \frac{e^T y}{m} + \frac{e^T z}{k} \\ &-A\omega + e\gamma + e \leq y \\ &B\omega - e\gamma + e \leq z \\ &y \geq 0, z \geq 0 \end{aligned} \tag{8}$$

Recently, the LP framework has been extended to cope with the feature selection problem [15]. In our research, we adopt formulation (8) as a classifier to minimize wrong classification.

Since formulation (8) is only used for separating two sets points, a seven-expression classification problem is decomposed to 21 2-class classification problems. In the training stage, 21 classifiers according to 21 expression pairs are formed with 21 pairs of $\{\omega, \gamma\}$. In the testing stage, feature vector of a testing sample is imported into these classifiers for comparisons. Fig.7 shows the classification result for original image in Fig.1 (a).

Fig. 7. Classification result

5 Evaluations

In our research, a commercial digital camcorder is connected to a computer for images acquisition and the system operates at about 20 frames/second in 320×240 images on a 3GHz Pentium V. Fig.8 shows the output of the system for a test video in which the subject poses a series of facial expressions.

Fig. 8. Examples of correct recognition

The recognition performance of our system is tested as follows:

1) **Person-dependent recognition:** In the training stage, every one of ten individuals is required to pose seven basic expressions in from of a camcorder. Then some frames are selected from the video stream to produce expression template for this person. In the recognition stage, these individuals pose expressions again and the system recognizes them and displays the result in time. To evaluate the recognition performance, the system also save the original video stream and recognition results. When the system ends its work, each individual are asked to label his expressions in the original image sequences. Results of the system are compared to the labels and then we have the recognition rate. The average recognition accuracy is 91% (See table 1).

Table 1. Person-dependent Recognition result

Expressions	Recognizing rate
Anger	91%
Disgust	86%
Fear	82%
Happiness	99%
Neural	93%
Sadness	91%
Surprise	97%
Average	91%

2) **Person-independent recognition:** The procedure is similar to that in 1). The difference is that expressions of seven individuals are used for training and those of other three persons are used for testing. One expert who is familiar with the seven basic expressions is asked to labels the testing video streams. Results of our system are compared to the labels and then we have an average recognition rate of 78% (See table 2).

Table 2. Person-independent Recognition result

Expressions	Recognizing rate
Anger	75%
Disgust	68%
Fear	65%
Happiness	89%
Neural	78%
Sadness	81%
Surprise	87%
Average	78%

6 Conclusions

Real-time and fully automatic facial expressions recognition is one of the challenging tasks in face analysis. This paper presents a novel real time system for expression recognition. The face pre-processing is implemented based on the skin detection and face geometrical structure, which can assure correct eyes detection under large illumination changes and some degree of head movement. The Local Binary Patterns operator is used here to describe face efficiently for expression recognition. The features detection procedure is insensitive to grey level changes. The holistic features also make the proposed system insensitive to some range of head movement. At last, 21 classifiers are produced based on linear programming technique and classification is implemented with a binary tree tournament scheme, which can minimize wrong classification. The system requires no special working conditions. Besides this, experimental results demonstrate that the system performs well less constraint conditions, even in some degree of illumination changes and head movement.

Acknowledgement

The authors thank CIMO of Finland and the China Scholarship Council for their financial support for this research work. The "Talent Training Plan" of the Northwestern Polytechnic University also provides financial support to this work and should also be greatly acknowledged.

References

1. P. Michel and R. E. Kaliouby: Real time facial expression recognition in video using support vector machines, Proceedings of the 5th International Conference on Multimodal Interfaces, (2003) 258-264
2. I. Kotsia and I. Pitas: Real time facial expression recognition from image sequences using support vector machines, Proceedings of Visual Communication and Image Processing, (2005), in press
3. K. Anderson and P. w. Mcowan: Real-time emotion recognition using biologically inspired models, Proceedings of 4th International Conference on Audio- and Video-Based Biometric Person Authentication (2003) 119-127
4. H. Park and J. Park: Analysis and recognition of facial expression based on point-wise motion energy, Proceedings of Image Analysis and Recognition (2004) 700-708
5. X. Zhou, X. Huang, B. Xu and Y. Wang: Real time facial expression recognition based on boosted embedded hidden Markov model, Proceedings of the Third International Conference on Image and Graphics (2004), 290-293
6. M. Pantic, Leon J.M. Rothkrantz: Automatic analysis of facial expressions: the state of the art, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22 (2000) 1424-1445
7. B. Fasel and J. Luettin: Automatic facial expression analysis: A survey, Pattern Recognition, Vol. 36 (2003) 259-275
8. Li.Stan Z & K.Anil, Handbook of face recognition, Springer-Verlag, 2004.9
9. B. Martinkauppi, Face color under varying illumination-analysis and applications, Dr.tech Dissertation, University of Oulu, Finland, 2002
10. J.Hannuksela: Facial feature based head tracking and pose estimation, Department of Electrical and Information Engineering, University of Oulu, Finland, 2003
11. T. Ojala, M. Pietikäinen, T. Mäenpää: Multiresolution grey-scale and rotation invariant texture classification with Local Binary Patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24(2002) 971-987
12. T. Ahonen, A. Hadid and M. Pietikäinen, Face recognition with local binary patterns. The 8th European Conference on Computer Vision (2004), 469-481
13. T. Ojala, M. Pietikäinen and T. Mäenpää. Multiresolution grey-scale and rotation invariant texture classification with Local Binary Patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence Vol.24 (2002): 971-987
14. K. P. Bennett and O. L. Mangasarian, Robust linear programming discrimination of two linearly inseparable sets, Optimization Methods and software, vol.1 (1992), 23-34
15. P. S. Bradley and O. L. Mangasarian, Feature selection via concave minimization and support vector machines, Proceedings OF THE 5th International Conference on Machine Learning (1998) 82-90